REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
07/30/2014	Final Technical	02/23/10-10/31/13

4. TITLE AND SUBTITLE

Simulation Methods for Design of Networked Power Electronics and Information Systems

5a. CONTRACT NUMBER

5b. GRANT NUMBER
N00014-10-1-0625

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Roger A. Dougal

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of South Carolina
Research Foundation
901 Sumter Street Suite 501
Columbia, SC 29208-0001

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)
ONR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This research accomplished four major goals related to tools for design, validation, and operation of future electric ship systems: 1) it developed multi-rate simulation in arbitrary mixed execution environments including tools to optimally configure bi-rate simulation models while considering step size and simulation stability, 2) it developed a new solver that exploits both latency methods and discrete event methods, 3) it developed and demonstrated methods for deeply integrating mature (commercial) simulation software into the VTB framework, and 4) it developed methods to support robust execution of simulations in a distributed computing environment and demonstrated the approach on digital signal processor (DSP) and field programmable gate array (FPGA) systems, and 5) it explored methods to account for the effect on system performance of the coupled networked flows of power and information.

15. SUBJECT TERMS
Simulation methods, latency insertion method, Simulation speedup

16. SECURITY CLASSIFICATION OF: U			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER *(include area code)*

Final Technical Report
Simulation Methods for Design of Networked Power Electronics and Information Systems

GRANT NUMBER N00014-10-1-0625

Submitted to

Peter Cho
Office of Naval Research
ONR Code 331
875 North Randolph St, Suite 1425
Arlington, VA 22203-1995

July 2014

Submitted By

Dr. Roger A. Dougal, PI
Dept. of Electrical Engineering
University of South Carolina

With contributions by PIs:

Roy Crosbie –California State University/Chico,
Chika Nwankpa - Drexel University
John Pearce - ISIM,
Antonello Monti - RWTH/Aachen

Technical POC
Dr. Roger Dougal
Dept. of Electrical Engineering
University of South Carolina
Columbia, SC 29208
Telephone: (803) 777-7890
Fax: (803) 777-8045
email: dougal@engr.sc.edu

Administrative POC
Danielle McElwain
Sponsored Awards Management
901 Sumter Street
Columbia, SC 29208
Telephone: (803) 777-1119
Fax: (803) 777-1119
email: dmcelwai@mailbox.sc.edu

Award Information

Award Number	N00014-10-1-0625
Title of Research	Simulation Methods for Design of Networked Power Electronics and Information Systems
Principal Investigator	Dr. Roger A. Dougal
Organization	University of South Carolina

Abstract

This research accomplished four major goals related to tools for design, validation, and operation of future electric ship systems: 1) it developed multi-rate simulation in arbitrary mixed execution environments including tools to optimally configure bi-rate simulation models while considering step size and simulation stability, 2) it developed a new solver that exploits both latency methods and discrete event methods, 3) it developed and demonstrated methods for deeply integrating mature (commercial) simulation software into the VTB framework, and 4) it developed methods to support robust execution of simulations in a distributed computing environment and demonstrated the approach on digital signal processor (DSP) and field programmable gate array (FPGA) systems, and 5) it explored methods to account for the effect on system performance of the coupled networked flows of power and information.

Two software tools that support the use of bi-rate simulations of linear systems were developed and documented. These are bi-rate versions of the single step-size optimization tools that were delivered last year. The first tool determines the stability of the system in the form of a root locus, and the second is a tool for selecting the largest possible simulation integration step sizes subject to accuracy constraints. Both tools allow the user to select from several numerical integration methods.

A new approach to simulation of dynamic energy-conserving networks was developed. The utility of this new solution method is especially noticeable when simulating large and complex systems such as the multi-disciplinary power systems of a naval combatant. The solver integrates our earlier work with the Latency Insertion Method (LIM) and our more recent explorations of Discrete Event Systems (DEVS) methods. The result is a revolutionary new high-performance, highly-parallelizable, multi-rate, time-domain simulation framework. Notably, it does not require time-consuming inversion of a large system matrix, nor, during model formulation, the tedious steps of algorithm-dependent discretization of the differential algebraic equations that describe the system.

We demonstrated extensibility and generality of the Virtual Test Bed (VTB) framework to support multiple solvers and their associated model libraries by integrating two quite different modelling techniques and philosophies -- ESL and VTB. This delivered a capability to build and run ESL simulations from within VTB while maintaining the look and feel of ESL. This led to new understandings of the requirements for integrating a first simulation tool and its model library into another second tool so that the first can be used natively within the second, and verified that VTB meets these requirements.

A new method for the simulation of nonlinear power networks was developed to overcome the main limitations in handling nonlinear and time-varying components. This second method operates within the framework of the Resistive Companion Method (the default solver

for VTB) by representing each nonlinear element of the network as a current injection source for which the state equations have been discretizing according to an explicit integration algorithm. The system model can be advanced using a fixed time step. The mathematical formulation of the method was completed and then implemented by using the code generating capability built into the desktop version of VTB to generate the actual C-Code solver that was demonstrated on a DSP cluster. A similar too was developed to generate the C-Code for simulation on FPGAs.

Methods were explored to account for the combined effects of power flows and information flows on the performance of a system. Two efforts to address this issue were pursued: one based on hardware and one based on software. Using a hardware test bed we characterized the effects of measurement delay errors (MDEs) on power flows in an "information embedded power system". Using a software test bed, we explored the causes of disturbances in the energy system and its embedded information system and developed new, expanded definitions (both by quality and quantity) of disturbances in terms of measurable characteristics of the information-embedded technology. Inferences will later be drawn from results of the two efforts for eventual verification/validation work.

Taken together, all of these methods permit accessing different simulation tools while within a common environment, they provide a capability to run multi-solver simulations (resistive companion – signal flow – state-space), and to compute different parts of a system model by using the most appropriate method, thereby making for more efficient simulation and, ultimately, superior electric ships.

Contents

Objectives

The overarching objective of this program is to develop methods for fast simulation of power electronic systems that are integrated with and controlled through data networks, and to instantiate examples of those methods in the VTB system and in other simulation tools so as to better support development of highly-automated and highly-efficient electric ships. The following specific sub-goals were defined to lead to the high-level program objective:

- *Multi-rate simulation in arbitrary mixed execution environments* – extending the existing case-by-case demonstrations of multi-rate simulations to a general method that enforces physics-based natural coupling between arbitrary solvers by developing algorithms, including anti-aliasing algorithms that support natural coupling of computed quantities, and by developing standards for the physical interface between solvers. **[California State University, Chico, University of South Carolina]**
- *Formal methods to support robust execution on a distributed environment* – exchanging of a small data sets at very high rates. For this purpose, typical computer clusters are not suitable because they introduce a large latency in the calculation. **[RWTH Aachen University, University of South Carolina]**
- *Rapid deployment of the system model to FPGA execution environment* – extending the capability for automatic generation of C code from the system definition to permit automatic generation of VHDL code. **[California State University, Chico]**
- *Methods for graphical editor to segregate the components of a system into separate parts that can be deployed as per the first and second bullets* – develop methods to permit use of a rubber-band cursor to select, from a large system diagram, a group of components that will be simulated on one environment, and other groups of components that will be simulated on other environments. **[University of South Carolina]**
- *Integration of the multirate ESL solver into VTB* – making ESL re-entrant so as to permit multirate ESL models to work within the VTB environment. **[California State University, Chico, University of South Carolina]**
- *Develop a low-cost method for DSP-based simulation and control* based on the use of arrays of processors by exploiting the code-generation framework of VTB. **[RWTH Aachen University]**
- *Develop methods to account for the combined effects of power flow and information flow on the performance of a system.* This includes both hardware approaches and software approaches. **[RWTH Aachen University, Drexel University]**

Accomplishments

The main technical accomplishments are briefly summarized here. Further technical details are provided in the related publications, which are described later in this report.

Multi-rate simulation in arbitrary mixed execution environments

Two software design tools intended for use with bi-rate linear simulations were developed and documented. The first characterizes the stability of a bi-rate simulation model in the form of a root locus, and the second identifies the largest allowable simulation (integration) step sizes subject to accuracy constraints. Both tools allow the user to select one of several numerical integration methods. A next phase of this work should investigate tri-rate systems. An alternate approach to multi-rate simulation using a similarity transformation was also developed, and the software was tested for a bi-rate system.

The single step-size version of the root locus tool uses Mathematica to convert a VTB schematic capture model into a state model and the root locus tool is then applied to the state model. Future work will extend the set of Chico software tools to include the VTB schematic capture and state-space matrices calculator with the other Matlab tools.

Formal methods to support robust execution on a distributed environment, and Methods to segregate components of a system model into separate parts

Coupling Method Assessment

A new method for the simulation of nonlinear power networks within the framework of the Resistive Companion Method has been developed. To overcome the main limitations of the Resistive Companion scheme in handling nonlinear and time-varying components, we describe a new method that uses a fixed simulation time step, while representing each nonlinear element of the network through a current source, discretizing its state equations with an explicit integration algorithm.

The proposed method can be executed on a multicore machine developed at RWTH Aachen University, based on intensive use of Digital Signal Processors (DSPs) for computational purposes, and on Field Programmable Gate Arrays (FPGAs) for signal routing. Herein, we refer to the hardware simulator as "DSP cluster".

The algorithm of the new method can be summarized in the following steps: describe the network, identify and substitute the nonlinear components in the network with a current or voltage source (the dynamics of which are described by state equations, which are discretized by using an explicit integration method), transform the now linear network into its associated discrete resistive network, and solve the resistive companion equations in parallel among a set of DSPs. Finally, the network solution is computed by forward and backward substitution in a single DSP. In case the resulting network is still too large for a single DSP, diakoptics can be used to further parallelize the execution.

Among the main advantages of the described method are the following:

- The network equations are formulated only once for a nonlinear power system.
- As the elements of the conductance matrix are dependent on the time step only, then triangular factorization can be performed offline (before starting the simulation).
- Each individual component represents a subsystem that can be solved independently of the rest of the system; this allows parallel computation of the solution among the set of DSPs.

- The size of the matrix that describes the whole system is reduced, since components solved using state equations do not present internal nodes.
- No multiple iterations (for convergence) are required during a single time step.

Before describing details of the simulator's structure, we define some of the relevant terms as follows:

A task is a set of commands for calculating the solution of one component of the network. These tasks can be divided into offline and online tasks. By offline we are referring to those steps of the algorithm that are performed before running the simulation, so they belong to the algorithm of our method, but they are not counted by the simulation clock; by online, we refer to those tasks that are performed in real-time, through the simulation platform.

The distribution of tasks over the hardware structure (*n* DSPs) is performed under one Master and many Slave processors. Given the above specifications, we now describe the conceptual steps of the method that we developed and investigated:

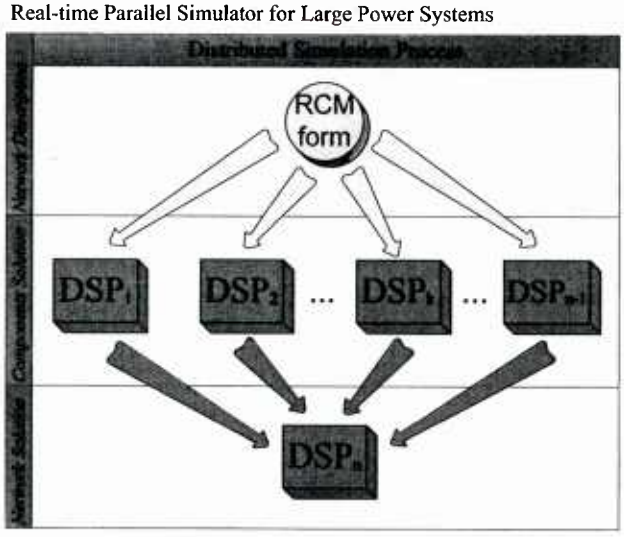

Figure 1 Phases of the Simulation approach process

Offline

The "Network Description" phase of the Simulation process (see Figure 1) is performed offline, and it executes the following sub-processes:

- Description of the network.
- Representation of each nonlinear component as a current source and substitution in the network description. The current source is controlled by the nonlinear component state equations, which are discretized using the 2nd order Adams-Bashforth integration algorithm:

$$x_{n+1} = x_n + \frac{h}{2}(3\dot{x}_n - \dot{x}_{n-1}), \qquad (1)$$

where h is the time step, and x are the state variables. This representation transforms the original network into a linear system.

- Representation of the now linear network using equivalent companion models. We substitute all the components originally linear with their discrete model, using the trapezoidal integration method:

$$x_{n+1} = x_n + \frac{h}{2}(\dot{x}_n - \dot{x}_{n+1}), \qquad (2)$$

and the equivalent current/voltage sources are introduced for the nonlinear components with their equivalent resistive companion models.

- Construction of the nodal equations with all the network components represented by their respective Norton/Thévenin equivalents; we thus obtain a problem in the form of linear algebraic equations, describing the state of the system at time t:

$$Gv(t) = i(t) - I, \qquad (3)$$

where G is the nodal conductance matrix, $v(t)$ the vector of node voltages, $i(t)$ is the vector of injected node currents, and I is the vector of known equivalent current sources.

- As the time step is fixed, then the conductance matrix G remains constant, and we can factorize it as $G = LU$, which transforms equation (3) into:

$$\begin{cases} Ly(t) = i(t) - I \\ Uv(t) = y(t) \end{cases}, \qquad (4)$$

Notice that only the right hand side of our nodal equation needs to be updated.

Online

The phases "Components Solution" and "Network Solution" of the Simulation process (see Figure 1) are performed online, and they execute the following tasks:

- The master DSP solves the system of linear equations in (4), by performing forward backward substitution and yielding the solution vector v.
- The master DSP broadcasts the solution vector of nodal voltages to the slave DSPs.
- The slave DSPs solve the internal equations of the nonlinear components associated with the current sources in (1).
- The slave DSPs send their solutions to the master processor, which updates the right hand side of (3).

These tasks are iterated for each step.

Mathematical formalization of the selected method

We know from the literature that, with state space analysis, any power system can be represented by a set of differential equations of the form:

$$\begin{cases} \dot{x} = f(x, u, \dot{u}) \\ y = g(x, u, \dot{u}) \end{cases}, \qquad (1)$$

where $x \in \mathbb{C}^{s_L + s_{NL}} \equiv \mathbb{C}^s$ represents the state variables, u the input vector, and y is the output vector. We are interested now in studying the stability properties of the numerical methods used for the integration of the differential equation of (1).

In order to investigate the stability of the simulation approach that we proposed, the differential equations of the power system under analysis were formulated as a partitioned system, where one of the sub-problems comprises the nonlinear part of the original problem, and the other sub-problem represents the linear part of the system. In this way, the linear sub-problem is discretized with the Trapezoidal rule (an implicit method), while the nonlinear sub-system is discretized with the 2nd order Adams-Bashforth integration method (explicit); this is called a Linear Multistep Compound (LMC) method of order $k = 2$. It is important to highlight that both sub-systems are integrated with the same fixed stepsize.

Formally, we can reformulate our problem as:

$$\begin{cases} \dot{x}^L = f^L(x^L, x^{NL}, u, \dot{u}) \\ \dot{x}^{NL} = f^{NL}(x^L, x^{NL}, u, \dot{u}) \end{cases}, \quad (2)$$

where $x^L \in \mathbb{C}^{S_L}, x^{NL} \in \mathbb{C}^{S_{NL}}$, and the superscripts denote linear (L) and nonlinear (NL). The discretized version of the state equation (1) can be expressed then as follows:

$$\begin{cases} x_{n+1}^L = x_n^L + \frac{h}{2}[f^L(x_n^L, x_n^{NL}) + f^L(x_{n+1}^L, x_{n+1}^{NL})] \\ x_{n+1}^{NL} = x_n^L + \frac{h}{2}[3f^{NL}(x_n^L, x_n^{NL}) - f^{NL}(x_{n-1}^L, x_{n-1}^{NL})] \end{cases}, \quad (3)$$

where h denotes the discretization time step.

Let us consider then the two following linear multistep methods:

$$\begin{array}{cc} Trapezoidal\ Rule & 2-step\ Adams-Bashforth \\ \rho_1(x) = (x-1) & \rho_2(x) = x(x-1) \\ \sigma_1(x) = \frac{1}{2}(x+1) & \sigma_2(x) = \frac{1}{2}(3x-1) \end{array}$$

Notice that these generating polynomials do not have in common any other root than 1, which satisfies the stability condition for partitioned linear multistep methods. We have then the following matrices that correspond to the partitioning of the differential system given in (2):

$$\begin{cases} \mathrm{P}(x) = \begin{bmatrix} x\rho_1(x)I_1 & 0 \\ 0 & \rho_2(x)I_2 \end{bmatrix} \\ \Sigma(x) = \begin{bmatrix} x\sigma_1(x) & 0 \\ 0 & \sigma_2(x) \end{bmatrix} \end{cases}, \quad (4)$$

where I_1 and I_2 are $S_L \times S_L$ and $S_{NL} \times S_{NL}$ unit matrices, respectively. These partitioning matrices (P, Σ) define our LCM discretization as:

$$P(x)x_{n+1} = h\Sigma(x)f(x_n), \quad (5)$$

with $h > 0$. Let S_i, with $i = 1,2$, be the stability regions of the methods (ρ_i, σ_i). Let us consider know the sets defined by:

$$D_i := D(a_i, b_i, c_i, d_i) = \left\{ q \in \mathbb{C}: Re\left(\frac{a_i q + b_i}{c_i q + d_i}\right) \leq 0 \right\},$$

where $a_i, b_i, c_i, d_i \in \mathbb{C}$, such that $a_i d_i - b_i c_i \neq 0$, for $i = 1,2$. Then, we know from literature that the set $D_1 = D(1,0,0,2)$ is identical to S_1. We also know from literature that if we use the Forward Euler method as a starting procedure for the 2-step Adams-Bashforth algorithm, then there exists a stepsize-coefficient $\gamma_0 = 4/9$ for boundedness and monotonicity of the method. The set $D_2 = \gamma_0 D(1,0,1,2)$ is contained in the stability regionS_2. With these sets, we can now define the following matrices:

$$A = \begin{bmatrix} I_1 & 0 \\ 0 & \frac{4}{9}I_2 \end{bmatrix}, \qquad B = 0,$$

$$C = \begin{bmatrix} 0 & 0 \\ 0 & \frac{4}{9}I_2 \end{bmatrix}, \qquad D = \begin{bmatrix} 2I_1 & 0 \\ 0 & \frac{8}{9}I_2 \end{bmatrix},$$

which define the following partitioning for our LMC method (P, Σ) given in (5):

$$\begin{cases} \tilde{\mathrm{P}}(x) = A\mathrm{P}(x) + B\Sigma(x) \\ \tilde{\Sigma}(x) = C\mathrm{P}(x) + D\Sigma(x) \end{cases}. \tag{6}$$

Then, the LMC method $(\tilde{\mathrm{P}}, \tilde{\Sigma})$ defined by (4) and (6) is A-stable.

For our LMC method (P, Σ), the linear test equation for studying the nonlinear stability analysis is $\dot{x} = Jx$, where the $s \times s$ matrix J is partitioned according to:

$$J = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix}. \tag{7}$$

We choose the inner product $\langle x, x \rangle = x^* x$, where the superscript $*$ denotes conjugation and transposition.

We define the matrix $K = J^* C J$:

$$K = \begin{bmatrix} J_{11}^* & J_{21}^* \\ J_{12}^* & J_{22}^* \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & \frac{4}{9}I_2 \end{bmatrix} \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix} = \frac{4}{9} \begin{bmatrix} J_{21}^* J_{21} & J_{21}^* J_{22} \\ J_{22}^* J_{21} & J_{22}^* J_{22} \end{bmatrix}.$$

Now, by (13), we have that:

$$\mu_E \left[AJD + \frac{h}{2} AK \right] \leq 0 \tag{8}$$

is sufficient for stability, where μ_E is the logarithmic norm based on the Euclidean norm; i.e., $\mu_E(M)$ is the maximum eigenvalue of $\frac{1}{2}(M + M^*)$. Now, using some properties of the norm on the left-hand side of (8), we find that the results of the simulation with our LMC method will be stable if we choose:

$$h \leq -2 \frac{\mu_E[AJD]}{\mu_E[AK]} \tag{9}$$

An important observation that needs to be pointed out is that, even when (9) gives a rather pessimistic estimate of the largest possible time step h for the simulation, the stability properties of the presented LMC method are better than those of the "worst" integration algorithm of our approach, which in this case is the 2nd order Adams-Bashforth.

Defining a system architecture independently of its simulation model

The objective of this task was to define and prototype a new method for capturing at a high level the architecture of a system (generally a power system) so that appropriate pieces of the system model could be assigned, via appropriate data translators, to different simulation tools. The highest level definition of the system, abstracted from any particular simulation environment, is called the "design space model". The developed translators convert and export design space data so that it can be consumed by other design or simulation tools. This permits all of the design space data and analysis data to be kept in one place and in one format, and it eliminates the time-consuming problem of having to manually translate design space data into the different formats that are needed by different analysis or simulation tools.

The architecture we implemented allows a design space model to be exported to external simulation tools via application program interfaces (APIs) in those tools, so that a human user does not have to interact directly with those tools. Instead, the human user simply specifies the types of analyses they wish to run on a design space model, then the required simulations are generated and executed, and the results are returned to the design space modeling environment for review by the user (Figure 1).

We demonstrated this capability by using a design space suitable for configuring and evaluating electric ship concepts. Design spaces for other types of systems may not have the same common features, but the template methods will be similar.

PIE solver as a method to accomplish multi-rate simulation

(This task originated under ONR grant N00014-08-1-0080 and transitioned to and was completed under N00014-10-10806)

We developed a new approach to simulating the dynamic performance of multidisciplinary energy-conserving networks. We call the new solver the Power Injection Event solver (PIE solver). One advantage of this new solution method is especially noticeable when simulating large and complex systems such as the power systems of a naval combatant. The PIE solver integrates our earlier work on latency methods (LIM)[1] with our more recent explorations of Discrete Event Systems (DEVS)[2] methods. The result is a revolutionary new high-performance, highly-parallelizable, multi-rate, time-domain simulation framework. Notably, the PIE solver does not require time-consuming inversion of a large system matrix, nor, during model formulation, the tedious steps of algorithm-dependent discretization of the differential algebraic equations that describe the system. (This discretization step also severely inhibits maintenance and upgrading of models). Instead models are advanced through time by a semi-implicit method. This approach permits expression of the model dynamics in natural mathematical forms (e.g. as differential algebraic equations). Event-based stepping (rather than fixed time stepping) of the PIE solver, depending on quantization levels and any inserted latency, permits rapid time advance during quiescent periods. Insertion of latency in every branch and at every node permits the system model to be efficiently distributed across many separate computing cores. An invention disclosure was filed for this new solver.

The PIE simulation method applies to a network model, where the network connects nodes. Every node has an "across value" representing e.g. voltage in an electrical circuit or velocity in a mechanical system. The across value of a node is its value relative to a reference point, such as ground in an electrical circuit or a point at rest in a mechanical system. Nodes are

connected to each other by branches, which are characterized by "through values" (the equivalent of current in an electrical circuit, or force in a mechanical system). Nodes inject across values to the branches, and the branches inject through values to the nodes. Most disciplinary frameworks follow the convention that the product of the through and across variables is power. Conservation of energy demands that the sum of all through variables at any node be zero.

The system equations are decoupled using latency methods, which requires either the exploitation of existing system latency, or the insertion of latency if none explicitly exists at each relevant node or branch (capacitors and mass are examples of across latencies in the electrical and mechanical disciplines, respectively; while inductors and springs are examples of through latencies in those same disciplines). The leapfrog integration method, as typically used in the latency insertion method (LIM), is replaced with a discrete event systems (DEVS)-based implementation in which state quantities are discretized rather than time quantities. The resulting equations are not solved using a constant time-step h, as is usual in the resistive companion approach. The result is an event-based simulation model having multi-rates. Update computations occur only when events occur, such as a transition to a new quantized state[3].

A prototype implementation of the PIE solver was developed using the Virtual Test Bed (VTB) framework. This implementation was used to explore the challenges in generalizing the method and to quantify the computational benefits for realistic simulation scenarios.

[1] J. E. Schutt-Ainé, "Latency Insertion Method (LIM) for the Fast Transient Simulation of Large Networks," *IEEE Trans. Circuits Syst. Fundam. Theory Appl.*, vol. 48, no. 1, pp. 81–89, Jan. 2001.

[2] E. Kofman and S. Junco, "Quantized-state systems: a DEVS Approach for continuous system simulation," *Trans Soc Comput Simul Int*, vol. 18, no. 3, pp. 123–132, Sep. 2001.

[3] C. Mamai, R. A. Leonard, I. Kondratiev, and R. A. Dougal, "A Hybrid Quantized Discrete-Event Simulation Using the Properties of the Latency Insertion Method," presented at the Summer Simulation Multiconference, Genoa, Italy, 2012.

•Rapid deployment of a system model to an FPGA execution environment

Field Programmable Gate Arrays (FPGAs) offer a number of advantages over general purpose computers such as highly-parallel operations and low-latency data operations. Although the use of FPGAs in simulations was originally motivated by a need for high speed in real-time simulation of power-electronic systems, FPGAs can also play a part in improving the responsiveness of non-real-time simulations, both in terms of speeding up simulations and as cost-effective computing power for large and complex systems.

In this research our goal was to provide a means of converting system models defined by schematic capture into efficiently executable FPGA code. This will permit one or more FPGAs to compute a simulation model as part of a co-simulation with VTB. An early focus was on generating efficient FPGA code for solving linear differential equations. Starting from a VTB schematic of a linear electric circuit, a Mathematica program extracted information that was converted into a set of linear differential equations in symbolic form. These equations were then combined with a user-selected numerical integration algorithm to produce a set of difference equations. Numerical values of symbolic parameters were entered to produce a set of matrices that defined the math model to be computed. These matrices were loaded into a matrix equation

solver that was preprogrammed into the FPGA. This data-driven approach permitted "the FPGA program" to remain constant and only the data was changed in order to compute a different model.

The method is illustrated in Figure 2 and Figure 3. Figure 2 shows a VTB schematic of a converter filter circuit that was used in the Unmanned Underwater Vehicle simulation model. A Mathematica program generated the input that described the differential equations needed to simulate this circuit in the FPGA. This is shown in the screen shot in Figure 3.

Figure 2: Circuit schematic of output filter of a power converter, as used to demonstrate automatic generation of code for an FPGA

$$\begin{pmatrix} \frac{u1[t]}{Ls\,\sigma} - \gamma\, x1[t] + ita\,\beta\, x3[t] + np\,\beta\, x4[t]\, x5[t] \\ \frac{u2[t]}{Ls\,\sigma} - \gamma\, x2[t] + ita\,\beta\, x4[t] - np\,\beta\, x3[t]\, x5[t] \\ ita\, M\, x1[t] - ita\, x3[t] - np\, x4[t]\, x5[t] \\ ita\, M\, x2[t] - ita\, x4[t] + np\, x3[t]\, x5[t] \\ -\frac{Tmech[t]}{J} + \mu\,(x2[t]\, x3[t] - x1[t]\, x4[t]) - \frac{f\, x5[t]}{J} \end{pmatrix}$$

Figure 3: Screen Shot of Mathematica-Generated Input to FPGA Code Generator

In the final implementation, the FPGA could compute up to 16 first-order linear differential equations. This was not a fundamental limit of the configuration; the FPGA could support larger systems of equations. But there is a trade-off to be considered between speed, which is maximized by a fully-parallel implementation in the FPGA, and problem size, which can be increased by the use of more sequential operations.

While this approach worked very well for solving linear sub-systems, there remained the important question of how to handle non-linear mathematics in the general case. To explore this, some simple instances of non-linearities (e.g. in induction motor) were handled on an individual basis with custom code. It may later prove possible to produce generic code segments for some types of non-linearities that can be used in a data-driven fashion similar to the approach for linear equations, but a more comprehensive solution will require the development of libraries of common FPGA submodels that can be linked together in the same way that conventional VTB simulations are linked.

Integration of Multirate ESL solver into VTB

The major objective of this task was to investigate methods of closely integrating the ESL simulation tool with the VTB. The reasons for this were (1) to demonstrate the capability of the VTB to support external third-party solvers, (2) to make available to users of the VTB the strengths of ESL, in particular: its multi-rate capability and powerful discontinuity handling features and (3) to support multi-solver simulations combining ESL's solver with other solvers in VTB. This can be seen as a more general demonstration of the advantages to be gained from accessing a range of different simulation tools in a natural way from a common environment. An introduction to this part of the project and the software tools involved will be found in [15].

A new solver was developed for the VTB to provide access to the ESL software and it made available the following features:

Standard ESL Entities

A set of VTB entities corresponding to the standard ESL simulation elements was developed. These are displayed in the VTB Schematic Editor when the ESL Solver is selected as the active solver in the GUI (applying the Component Library filter). These entities can be interconnected to form one or more subsystems that will be solved using ESL's numerical integration methods. We also created a set of standard entities (cross-over entities) that permit connection of ESL components with components that use another VTB solver, thereby allowing construction of system models that rely on different solvers.

Ability to extend the standard set of ESL Entities

For the standard built-in entities, ESL code associated with an entity is identified by a special *EslIseClass* parameter which relates to a fragment of built-in XML code, which acts as a template. New ESL Entities may be created using VTB's Entity Designer tool by specifying ESL input and output ports. The associated XML code may be specified directly through an *EslEntityXml* parameter. Once a new Entity is created, it becomes immediately available in the Schematic Designer Component Library.

Ability to specify ESL Entity parameters and ESL Solver parameters

The ESL Solver provides access to ESL Entity parameters through VTB's Schematic Designer. Entity parameters may be changed at the beginning of a simulation run or (in some cases) when a run is paused. Access is also provided to particular ESL simulation parameters such as: choice of numerical integration algorithm; integration and discontinuity error tolerances; minimum number of integration steps per VTB time step, etc.

Interactive running of the simulation

When the user invokes a run of a simulation in the Schematic Designer, the ESL Solver first identifies any ESL Entities (whether built-in or extensions) and then accesses the corresponding XML fragments. Next, it analyses the entity, and creates a mapping between the attributes specified in the XML fragment and VTB parameter, and maps the input and output ports specified in the XML with VTB ports. ESL source code is then generated, making use of the templates in the XML fragments. An ESL utility is then invoked to create a .NET assembly which represents the sections of the VTB schematic that comprise ESL Entities. All of the above operations take place in the background, the only indication to the user is a short delay before the simulation starts.

When the VTB framework moves the simulation forward a step, the ESL Solver invokes the .NET assembly to advance the ESL code. This is achieved by obtaining the values of the ESL model variables that have been designated as needing to be set in a step and setting the corresponding VTB port values. Then it invokes the ESL model to run forward to the next ESL communication interval (corresponding to one VTB step). A more detailed explanation of these operations will be found in [11].

Modularization

Modularization within ESL sections of a VTB schematic can be achieved in three ways:

Subsystems: Since schematic diagrams comprising only ESL components are essentially no different from schematics of other types of VTB components, the VTB subsystem facility can be used to break ESL schematics into smaller units.

ESL Graphical Submodels: The VTB Module Designer can be used to create a module comprising ESL Entities in the normal manner. When a simulation is initiated from Schematic Designer, the ESL Solver identifies any ESL modules and maps these into ESL submodel code. Any instance of the module in a schematic is then mapped into an ESL submodel call. This procedure avoids duplication of code and allows such modules to be processed in an efficient manner by the ESL software.

ESL Textual Submodels: ESL language code can be included as part of a simulation by importing ESL textual submodels into a VTB schematic diagram. This is done with the tool – *ESL Submodel to VTB Entity* – provided with the ESL VTB Extension package. The tool allows the user to specify files containing ESL textual submodels and it automatically creates corresponding ESL Entities, which are immediately available from the Schematic Designer Component Library. The tool gives the user some scope to rename the entity and its input and output ports and further to specify which submodel inputs are to be treated as parameters in Schematic Designer. This *ESL Submodel to VTB Entity* facility is used in the current VTB 2012, and it replaces the *Embedded Segment integration* facility used in VTB 2009, as described in [11].

Further detail of modularization aspects will be found in [4].

Interconnection of ESL and non-ESL Entities

The ability to connect ESL sections of a schematic to non-ESL sections is provided through special cross-over entities which allow the connection of ESL ports to Signal ports and vice versa.

Delivery

ESL integration with the VTB is delivered through an *ESL VTB Extension* installer which installs the appropriate DLLs. A prerequisite is that a suitable licensed version of the ESL software is installed on the user's computer.

Low-cost method for DSP-based simulation and control

Development of the code generation capability

The simulation tool Virtual Test Bed (VTB) was used as a framework for implementing a new approach to executing simulation models on a DSP cluster. The new approach uses the code generation capability of VTB to create a C-Code version of the system model and solver that are then executed on the DSP Cluster.

We considered it important to be able to execute the method also when using the desktop version of VTB in order to give the user the possibility to evaluate, case by case, the effects of the new simulation method before deploying the code on a DSP cluster for execution in real-time.

From the user´s point of view this feature allows for reusability of the simulation schematic setup. In fact, once a system model is created by using the schematic editor of VTB, the user should be able to simulate that schematic by using either the traditional resistive companion solver or the new approach developed here.

An important reason to allow this capability is to provide the user with the possibility to evaluate on her/his own desktop the accuracy and stability that will be available when execution is performed on the DSP Cluster. Thus the purpose is not to speed up the desktop simulation. Hence, the implementation of this capability has been done without looking to parallel execution and to the optimization of the code execution but focusing only on all the aspects of the method that impact the stability and accuracy, while reusing as much as possible the actual structure of the VTB solver.

It was possible to implement such a capability by focusing on the single component models without modifying any aspect of the solver structure. Linear components do not need any modification while non-linear components required a new parameter, named "Integration", and a new method, named "Internal_Step". These two were defined.

The parameter "Integration" has two allowed values: "Implicit" and "Explicit". When the solver calls the "Step" method of the non-linear components first the parameter "Integration" is checked. If the value is set to "Implicit", the simulation is performed with the current solver of VTB, which follows the Resistive Companion method. If the parameter is set to "Explicit", the "Step" method calls the method "Internal_Step" where the state equations, which are internal to the components, are solved, based on the previous network solution. Then the "Step" method

presents as Jacobian the linear equivalent and as Equivalent vector the linear equivalent representations that have been updated with the values computed by the "Internal_Step".

Thanks to this approach it is possible to run, on the desktop version of VTB, both of the solution methods by just selecting a parameter for the non-linear components.

We now present a simple test case to show how different solution methods can be selected for each non-linear component, through use of the VTB GUI. The results obtained with the new method are compared against those obtained by using the traditional solver.

The test case, Figure 4, consists of a DC non-linear machine with power supplied though a buck converter. The DC link is powered from an ac source via a three-phase rectifier. A mechanical torque load with viscous friction loads the motor. The motor speed is controlled via a PID controller that acts on the duty cycle of the buck converter.

The buck converter is represented by an averaged linear model while for the diodes and the DC machine nonlinear models are considered.

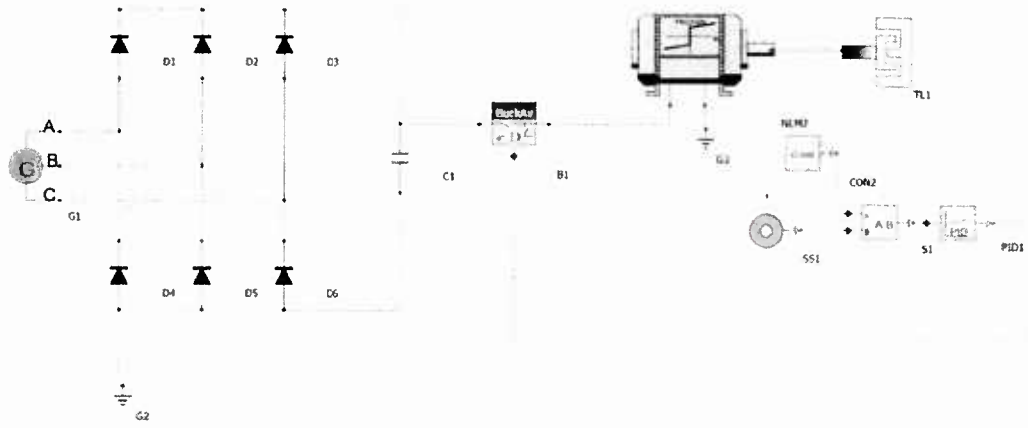

Figure 4 Test case

Figure 5 shows a screenshot of the VTB interface to set up component parameters: it is possible to see how, taking as example the non-linear DC machine, the integration method can be selected between "Implicit" and "Explicit". It is important to recall that the selection of the "Implicit" option means that the component equations are integrated using an implicit method (Gear's) and so integrated in the system Jacobian. The selection of the "explicit" option, instead, implies that the non-linear component is substituted by its linearized version in the resistive companion still solved with an implicit method but the sources of the equivalent linear circuit are updated at every time step with the so called 'Internal_Step'. In the 'Internal_Step' the non-linear state equations of the components are integrated using an explicit integration method.

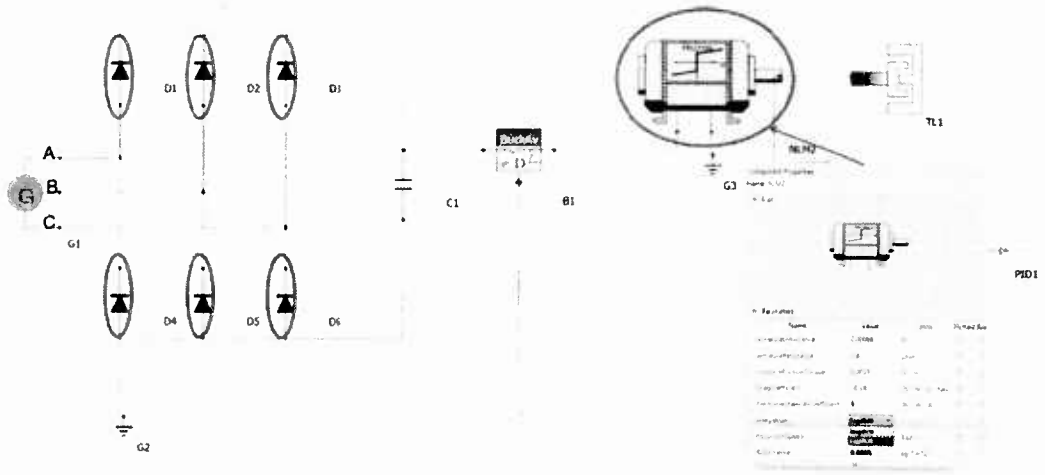

Figure 5 Integration method selection

The test case of Figure 4 was first executed using the traditional solver to create a base case. For this purpose the option "Implicit" was selected for each non-linear component. Then the option "Explicit" was chosen for the diodes and for the DC machine to evaluate the accuracy of the proposed method. In both cases a time step of 10μs was used.

Figure 6 shows the machine speed, during the first instants of the machine start up, as computed in the two cases. The difference between the numerical results in the two cases are summarized in TABLE 1, where the motor speed at the time instant 0.01s, is calculated according to the two methods. As shown in Figure 6, the error introduced by the new solution method is very small.

TABLE 1 TEST CASE RESULTS

	Speed (rad/s)
Implicit Solver Fig7.a	89.98
Explicit Solver Fig7.b	90.18

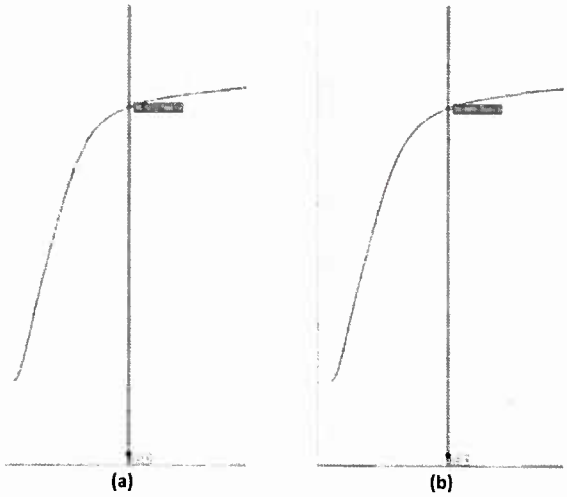

Figure 6 DC machine speed. (a) Implicit Solver (b) Explicit Solver

For the generation of code, different options are available from the combo box included in VTB GUI. Three main possibilities are of interest in this context:

- Standard C Code generation
  This option generates C-Code of the traditional resistive companion solver for the specific schematic.
- Explicit solver For Shared Memory Machine.
  This option generates C-Code for the execution of the new parallelization method on shared memory machines. OpenMP is used. We provided this option to allow for future development of the real time simulation execution on a PC Cluster.
- Explicit solver for DSP
  This option generates code to be executed on the DSP cluster. This solver is explained in detail next.

Development of the method for a DSP cluster

We now proceed describing the implementation of the software to be executed on the DSP cluster.

In Figure 7 a simplified scheme of the operation performed during a simulation step of VTB is shown. Looking at the "Natural Step" (first block of the second column, whose steps are blown up in the third column) we can see that five main steps are considered. Due to the nature of the method that we are proposing, which is linear (and we assume time invariant) from the resistive companion point of view, the steps named "Update system Jacobian" and "Check Convergence" do not need to be performed.

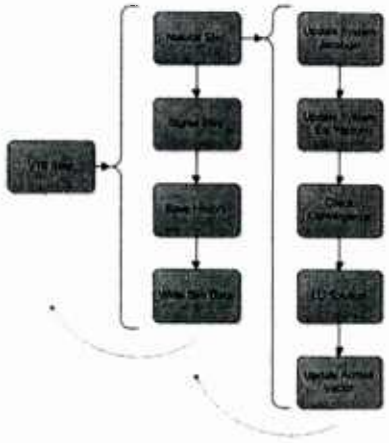

Figure 7 Standard VTB C-Code solver structure

Figure 8 shows a new solver structure that maximizes the benefit of the proposed method. As mentioned before, the LU decomposition is not performed online. Instead, the LU factorization matrices are stored and only the forward and backward substitution of (3) is performed online. A new stage named "Internal Step" has been introduced, in this phase the "Internal_Step" method of all the non-linear components is called, the "Internal_Step" method of each component behaves exactly like the one considered for the desktop implementation, the only difference in that now the "Internal_Step" of each components is directly called by the

solver and not by the "Step" method of the components. This modification makes it possible to parallelize execution of the "Internal_Step".

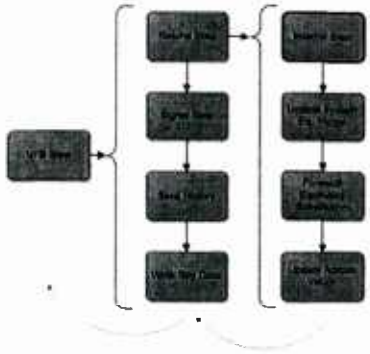

Figure 8 New VTB C-code solver structure

Figure 8 illustrates the structure of the software executed on the master DSP. Methods for parallel execution of the "Internal Step" among the different slave DSPs is still under development. At the same time thanks also to use of the first DSP cluster prototype, we are focusing on optimizing the software execution for a particular DSP while considering both performance and memory usage.

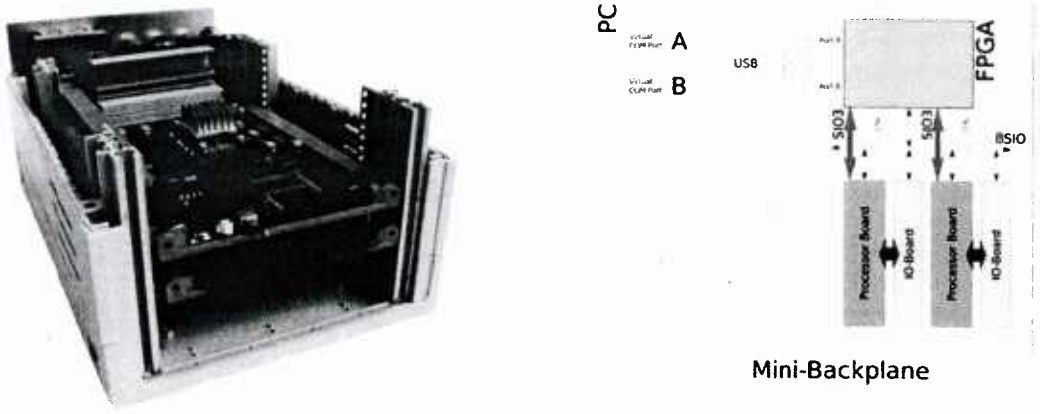

Figure 9 DSP Cluster Prototype

Rapid deployment of VTB system models to FPGA execution environments

Research contributions from CSU, Chico were focused in 3 areas: software tools, FPGA programming methods, and co-simulation using FPGA, ESL, and VTB.

The work on software tools continued to develop Matlab-based tools intended to assist simulation developers to produce accurate models and support their verification as part of the VV&A process. Work also continues on producing software support for developing FPGA-based models. FPGA programming is a labor-intensive, time-consuming, and error-prone process and the aim of the Chico research was to allow a developer of FPGA-based simulations to generate executable code from a schematic capture diagram of the system without the need to program the FPGA directly.

Demonstrations were developed that combine elements from different teams in the Simulation Methods research program. One such demonstration consisted of a co-simulation that

combined Chico FPGA software with the VTB in an application that uses FPGA, ESL and VTB models. Further details of that aspect of the research are described next.

Software Tools

Two software design tools intended for use with bi-rate linear simulations were completed and documented. These are bi-rate versions of the single step-size tools that were delivered in the penultimate project year. The first tool determined the stability of a system model in the form of a root locus, and the second tool identified the correct maximum integration step sizes for given accuracy constraints. Both tools allow the user to select one of several numerical integration methods.

The single step-size version of the root locus tool was recently demonstrated. Mechanistically, Mathematica was used to convert a model described by a VTB schematic into a state model and then the root locus tool was applied to that state model. Future work is planned to extend the set of Chico software tools to include the VTB schematic capture and state-space matrices calculator with the other Matlab tools.

•Methods to account for the combined effects of power flow and information flow on the performance of a system

Two distinct efforts addressed this issue: one based on hardware and a second based on software. Under the hardware effort, we enhanced an experimental platform that was previously created to characterize measurement delay errors (MDEs) in an "information embedded power system". Meanwhile, under the software effort, we developed definitions (both by quality and quantity) of disturbances with respect to energy system assessment which may be anticipated by measurable characteristics of the information-embedded technology. Specifically, causation of disturbances in the energy system and its embedded information system are highlighted in new, expanded definitions. It is anticipated that inferences will be drawn from results of the two efforts for eventual verification/validation work. More details are given next.

a) Hardware

Towards the enhancement of an experimental platform, the previously used 2 bus platform was replaced by a 3 bus platform. This improved depiction of actual conditions in the plant level of the information-embedded power system. Data was collected for the purpose of quantifying effects on delay of messaging and hence overall network performance in the face of constraints on communication bandwidth, congestion, and contention for communication resources. Studies were conducted into enhancing delay metrics. Additionally, with these metrics it is anticipated that delay compensation studies through characterizing the effect of inherent transmission traffic between the data collection system and the remote (monitored) data acquisition stations.

Both plant and instrumentation hardware were acquired to set up the 3 bus power system (a series connected transmission line + inverter + resistive load setup). Remote data acquisition and actuation were performed through the help of NI LabView software controlling 3 oscilloscopes positioned at the 3 buses. In addition, the study of network traffic communication bandwidth, congestion, and contention will be emulated through client/server interaction between computers acting as remote terminal units (RTUs). This study required client-side traffic shaping and monitoring software for these purposes.

For the purposes of validation and verification of this work, a benchmark power electronic converter system was selected. To quantify performance of these traffic measurement techniques, we have run analogous software experiments using this benchmark system.

We developed a model and simulation of an information embedded power electronic converter. Initial work concentrated on the generic issue of effect of delays on the overall analysis of the converter. From the corresponding results observation were made about measurement errors that result from delays in delivering the measurements. Network delay models in the form of exponential and logistic growth types were used to represent behavior of measured voltage and current values from a boost converter. This work differentiates itself from the OpNet work (see following software subsection) in that it is an attempt to develop a mathematically based representation of the overall system as opposed to directly simulating network components on an element by element basis (which OpNet S/W accomplishes).

b) Software

Measurement (or messaging) delay theory, modeling, and simulation are effectively influenced by underlying assumptions of actual communication messaging behavior. This is of course rooted in actual messaging rules between network resources. For example, it may be appropriate to view a typical RTU in a power grid as an event-driven device in the following sense. Assume an RTU measuring voltage values on a bus. Let V be the voltage reading at time t0, and let V' be the subsequent voltage measurement at time t1. The RTU will send an updated voltage reading to the control center only if $|V - V'| > \delta$, that is, subsequent voltage measurements differ from V by at least a user-defined threshold δ, and if the RTU continues to observe this difference over a time interval δt, also defined by the user. A major advantage of this event-driven scheme is that during normal and stable operation of the power grid with few voltage/current fluctuations, the messaging load imposed on the communication network is not very high.

The above-described scheme has a serious drawback in that it is susceptible to "event showers" during periods of abnormal grid operation. When the power grid is overloaded during hot summer days or experiences bus/generator failures, voltage and current values in some portions of the grid may fluctuate quite rapidly. Since the corresponding RTUs respond to these fluctuations (events), update messages are sent more frequently to the control center, thereby, imposing a very large messaging load on the underlying communication network.

We have developed both the theory and tools to detect and mitigate the above-described scenario by analyzing the network traffic, that is, messages injected by the RTUs into the network, for characteristics such as bandwidth consumption, message delays, and drop rates. Our hypothesis is that if one can learn the nominal network traffic injected by a grid that is functioning normally, then anomalies in the network-traffic flow can be quickly detected and used to localize potential problems in the grid.

We developed a simulation model of a large hierarchical network topology and conducted several experiments with different message injection rates by different flows of traffic. In our first set of experiments, we assumed Poisson traffic and simulated two states: one with each user injecting traffic into the network at a nominal rate and another in which two users enter an anomalous state and inject traffic at much higher rates. We then improved our simulation environment by adding realism to both the network topology and to the traffic pattern. Research over the last decade has established that a scale-free topology (one whose degree

distribution follows a power law) offers a better approximation of most real information networks including the Internet or a dedicated information network used in conjunction with the power grid. We developed a 300-node topology of a scale-free network for our delay experiments. We have developed a model of a self-similar packet injection process for our use in all of our simulation experiments. Our work so far is able to detect the presence of anomalous flows of traffic under these simulated conditions.

We have developed and validated a methodology to detect incipient faults in computing systems such as those used to manage the power grid. Incipient faults slowly deteriorate the system's performance over time and if left undetected, the end result is usually a complete system failure. The new method combines tools from information theory and statistics: entropy and principal component analysis (PCA). The entropy calculation summarizes the information content associated with the collected low-level metrics and reduces the computational burden incurred by the subsequent PCA step which detects underlying patterns and correlations present in the multivariate data, as well as distortions in the correlations indicative of an incipient fault. As a case study, we used the technique to detect memory bloat within selected software applications under dynamic operating conditions, showing that small leaks can be detected quickly and with a low false alarm rate. Our method is also robust to the periodic/seasonal patterns affecting the metrics used to detect the fault. For use in the general case, we are currently extending our case studies to be able to classify network traffic as well and to detect the presence of anomalies such as those caused by event showers discussed earlier.

List Publications (with Abstracts)

Publications by the project team are listed here, grouped according to the project goals that the publications addressed. The abstract from each publication is reproduced here to help clarify content.

•Multi-rate simulation in arbitrary mixed execution environments

[1] Bednar R. and R.E. Crosbie, "Multi-Rate Solution of Linear Differential Equations Using A Similarity Transformation" Proceedings of the 2012 International Conference on Grand Challenges in Modeling and Simulation, Genoa, Italy, July 8-11, 2012, SCS.

Starting with a system of linear differential equations in state variable form, this paper shows how a similarity transformation can be used to decouple the differential equations and put them into a form that allows separate numerical integration for each of the transformed first order differential equations. Thus a larger integration step size can be used for solving one or more of the transformed differential equations with slowly varying dynamics while a smaller step size can be used to solve the differential equations with rapidly changing dynamics. This method of multi-rate integration is not restricted to systems that are physically separable into subsystems with just fast or just slowly changing dynamics, or to bi-rate simulations. Two examples illustrate this method of multi-rate integration and compares results using a more conventional technique.

•Formal methods to support robust execution on a distributed environment

[2] Benigni, A. Monti, R. Dougal "Latency Based Approach to the Simulation of Large Power Electronics Systems" Early Access IEEE Transactions on Power Electronics 2013, DOI: 10.1109/TPEL.2013.2274175

We define a new hybrid simulation scheme that combines Latency Insertion with Nodal and State Space Methods. The conditions for stability and multi-rate time stepping of the hybrid method are defined. Further, we show how to define appropriate flow variable injections so that the hybrid method can be extended to applications involving either multiple technical domains or hardware in the simulation loop. The hybrid method permits significant increases of simulation speed while contributing less than 1% to RMS simulation error.

[3] Benigni, A., "Latency exploitation for parallelization of power sysetm simulation" Ph.D Dissertation 2013, ISBN978-3-942789-4

The realization of future power systems is expected to leverage on the exploitation of renewable sources and on an optimal use of the available energy at every instant in time. The practical implementation of these two concepts requires significant changes in the equipment and operation of power systems. All in all, these changes will increase complexity. In this scenario, the design of each part is a challenge because of the interactions and the dimension of the problem. In first place, the impact of the individual elements on the system cannot be inferred with analytical methods, instead numerical simulation is necessary. In second place, numerical simulation must be compatible with dynamic models, accommodate multi-physics and solve very large systems in a reasonable time, not to slow down excessively the design process and to possibly allow for real-time execution.

Most commonly used system simulators are based on nodal methods in electrical engineering and state space methods in mechanical engineering. Both approaches, if implicit integration is required, rely on the inversion of large matrices at each solution point. The matrix inversion computation time approximately scales with the cube of the matrix size, and strongly dominates the overall computation time, making the solution of large power systems unfeasible. To perform fast simulation of future power systems in a scalable way, with detailed representation of the single components, highly parallelizable simulation methods have to be developed, beyond what is already available in literature.

This dissertation focuses on the use of latency for parallelization of power system simulation. Exploiting the latency naturally introduced by reactive elements, two highly parallelizable and scalable simulation methods were developed. The proposed methods were then extended to multi-rate execution with further reduction of the execution time. This was achieved by exploiting the difference in the time constants of the various parts.

The first method based on the use of the Latency Insertion Method allows for decoupling the system in different parts, each then modeled and simulated using the classically most suitable simulators (e.g. nodal or state space based). This effective and scalable method is extremely flexible as it allows for different types of simulators to be interconnected. Because of this, though, the execution time can not be deterministically predicted, as a consequence this method may not be considered fit for real-time execution.

The second method exploits latency to couple nodal with state space methods. In a nodal simulation framework the nonlinear components are solved explicitly using state equations and then represented with a linear equivalent. This method yields a very high level of parallelization and a predictable execution time. Hence, this method is fit for real-time execution.

This dissertation is organized as follows. In chapter 1 an overview on the most common approaches to dynamic time domain modeling and simulation of electrical circuit is presented to create the fundamental background for what is presented in the rest of the dissertation. In chapter 2 the motivations for this work are presented. In the same chapter most significant and related works on latency exploitation for simulation purpose are presented together with a brief explanation of the proposed methods. In chapter 3 and 4 the two methods, i.e. Modified Latency Insertion Method and Latency Based Linear Multistep Compound Method, proposed is this dissertation are presented in detail. The accuracy and stability of the two methods is analyzed and exemplified with case studies. Multi-rate execution of the proposed methods is also presented and analyzed thanks to the use of examples. To conclude the extension of the two methods to multi-physic applications is illustrated. In chapter 5 simulation results from larger and non-linear test cases are presented. In the conclusions section the main characteristics and limits of the proposed methods are discussed.

[4] M. Marin, A. Benigni, A. Monti. "New Approach to Parallel Simulation of Large Power Systems," Proc. Grand Challenges in Modeling and Simulation, June 2011, The Hague, Netherlands, pp. 1-6.

The need for a fast numerical method that ensures stability and accuracy for real-time simulation of power systems still remains a challenging problem for researchers. In this paper, a new method based on an explicit integration method and parallel execution on DSPs is proposed. Some examples and preliminary results are included for illustrative purposes.

[5] Marin M., Collins P., Monti A. "Simulation Methods for Large Power Networks," Internship Report, Department of Knowledge Engineering, Maastricht University, The Netherlands, July 2012.

The real-time simulation of complex and large power systems remains as one of the biggest challenges for the electrical engineering community. The increase in size and complexity of such grids keeps asking for improvement in the development of new techniques, as well as software and hardware platforms, in order to obtain faster and more accurate and reliable representations of these networks.

The present work addresses the problem of simulating large dynamic nonlinear power systems in real-time, in a cluster of Digital Signal Processors (DSP). A new scheme is proposed for supporting the implementation of a parallel real-time simulator, based on a combination of the Resistive Companion Method (RCM) and the state equations approach. The conditions for guaranteeing a numerically stable simulation satisfying the time constraints are derived.

•Rapid deployment of the system model to FPGA execution environment

[6] Bednar, R., J. Zenor and R. Crosbie, "Simulation of a Squirrel-Cage Induction Motor: An Exercise in Modeling and Simulation" Proceedings of Conference on Grand Challenges in Modeling and Simulation, Toronto, Canada, July 2013, SCS Received Conference 2nd Best Paper Award

Stages in the development of a field-programmable gate array (FPGA) simulation of a squirrel-cage induction motor are described that clearly illustrate some of the problems

that can arise in developing simulations. The authors have experience in software engineering, computer architecture, numerical analysis, modeling and simulation, and a background in electrical engineering, but relatively little expertise in electrical machines beyond the basic electrical principles on which they are based. The authors do not intend to present state-of-the-art advances in induction motor models but rather to use their experiences to illustrate some of the problems that face developers and users of simulation in a more general sense.

The induction motor offers a good example for study in that the math model can take different forms depending on the nature of the application and that differences in basic assumptions, notation and sign conventions are common in the literature. The simulation itself also poses problems in terms of the choice of numerical integration algorithm and step-size selection. The need for care in the selection, interpretation and implementation of the appropriate model is clearly demonstrated. The "Grand Challenge" in this study is the ultimate one of using Modeling and Simulation effectively.

[7] Mish, S., J. Zenor and R. Crosbie, "An Efficient FPGA Matrix Multiplier for Linear System Simulation", Proceedings of Conference on Grand Challenges in Modeling and Simulation, Toronto, Canada, July 2013, SCS

FPGAs are increasingly being used as a key component in cost-effective high-performance simulations. Most of these efforts have been application specific requiring a custom design. Coding an FPGA can be very challenging but simple, non-optimal configurations are possible using graphical tools such as the Simulink blockset. Greater complexity and improved performance can be gained using a Hardware Description Language (HDL) such as VHDL or Verilog.

The problem of producing an FPGA-based general-purpose simulation system is considerably more demanding. The design of FPGA solutions that perform complex mathematical operations efficiently is difficult and compile times can be very long. One approach to providing this capability to simulation designers who are not expert FPGA programmers is to use pre-compiled FPGA structures driven by data representing a particular model.

A first step is described which provides for the solution of systems of linear differential equations. Non-linear features will be added later. The key to this first step is a flexible, efficient FPGA implementation of a matrix multiplier. An approach is described that improves on previously published methods; that can handle matrices of any size and up to over 100*100 on a single device; that offers flexibility in the choice of FPGA resources, such as DSP (digital signal processor) slices or LUTs (look-up tables); and is linearly scalable in its use of these resources.

The method is used as part of an implementation of a multi-rate benchmark representing an unmanned underwater vessel. The matrix multiplier is used in the simulation of the 3-phase induction motor drive of the vessel. Non-linear product terms are also handled in this model. Non-linear features will be added in future in the form of libraries of non-linear elements.

[8] Zenor, J., R. Bednar, R. Crosbie, D. Word, K. Kredo II, and N. Hingorani, "Using FPGAs for Low-Cost High-Performance Simulations", Proceedings of 2013 IEEE Electric Ship Technology Symposium,22-24 April 2013, Arlington VA.

With their flexible architecture capable of highly parallel processing, field-programmable gate arrays (FPGAs) offer exciting possibilities for supporting low-cost high performance simulations, both real-time and non-real-time. In effect the FPGA offers a way of customizing the processor architecture to match the structure of the simulation rather than the more familiar problem of making a program conform to the fixed processor architecture. Using FPGAs in this way, however, comes at a cost. FPGAs are not easy to program efficiently, especially in comparison with modern simulation software products such as, for example, Matlab/Simulink, SPICE and the Virtual Test Bed (VTB). Research at CSU, Chico aims to develop simulation systems that incorporate FPGAs while providing a programming interface that offers similar ease of use to such commercial simulation products. Efforts are also directed to the design of interfaces that can provide both the low latency and high transfer rates that are necessary.

[9] Zenor, J. J., R. Bednar, R. E. Crosbie and N. G. Hingorani, "Efficient Real-Time Simulation of Linear Differential Equations Arising From Simulation of Electronic Power Systems" GCMS 11 Conference, The Hague, Netherlands, June 27-30, 2011

Recent research has focused on developing techniques to support efficient, low-cost, high-speed, real-time simulation of modern power electronic systems. The higher speed components of these systems require frame times of a few microseconds or less. Digital Signal Processors (DSPs) and Field-programmable gate arrays (FPGAs) have been investigated for applicability to meet the computational requirements for the high-speed components in power systems. Though demonstrated to have the performance required, initially the implementation effort and development time required was excessive. First, techniques for solving the differential equations arising from the models of power system components were investigated that met the requirements of a high-speed real-time simulation: simplicity, predictability of computation time, stability, and accuracy. Methods and tools for quickly and automatically implementing the high-speed solutions to these equations in FPGAs were then developed. The FPGA based simulations were integrated into the Virtual Test Bench (VTB) for graphical presentation of the results, and to take advantage of the VTB model libraries and capabilities for rapid development of simulations for the lower-speed components of the power systems.

[10] Bednar, R. and R. E. Crosbie, "MATLAB Tools for Power Spectral Analysis, Simulation Step-Size Optimization, and Bi-Rate Root Locus Generation", GCMS 2011 Conference, The Hague, Netherlands, June 27-30, 2011.

This paper provides an overview of three Matlab tools that were originally developed for the simulation of power electronic systems, and have since been re-written for more general application. One of the programs can be used to determine power spectral density and total harmonic distortion of simulation outputs. It can be used to assist in the selection of appropriate integration step-size using frequency domain rather than the more usual time domain data. A second program provides a method of determining maximum allowable simulation step size using an optimization approach, and the third performs root-locus studies for bi-rate simulations. An example illustrating each tool is presented

[11] Justa, J., J. J. Zenor and K. Kredo II, "Efficient High-Speed Ethernet for Real-Time Simulation" GCMS 11 Conference, The Hague, Netherlands, June 27-30, 2011

In this paper, we introduce a method for interfacing a design space model with an analysis tool, a method which can be expanded to any number of tools. We created a software application for modeling an electric ship design space, and then implemented an interface between this model and a simulation tool. The interface is capable of taking the relevant data for a given analysis from the design space model and translating it into a format which the simulation tool can work with. The interface then runs the simulations, processes the data given back to it by the simulation tool, catalogs it, and lets the user access this information. As a result, all the design space and analysis data exists within one model in a format which can be automatically translated to associated simulation tools, which eliminates the need for design teams to manually translate data from one simulation tool to another. The result is that an engineer can operate solely within the confines of the design space modeling tool, analyzing the design space in various ways and critiquing the results without even having knowledge of the existence of the separate simulation tools.

Methods for graphical editor to segregate the components of a system into separate parts

[12] Conklin, G. Dougal, R.A. Langland, B.A., "Using Simulation Generation Templates to Interface Simulation Analysis Tools with Design Space Models," Proceedings of the 2012 International Conference on Grand Challenges in Modeling and Simulation, pp. 92-99, Genoa, Italy, July 8-11, 2012, SCS.

In this paper, we introduce a method for interfacing a design space model with an analysis tool, a method which can be expanded to any number of tools. We created a software application for modeling an electric ship design space, and then implemented an interface between this model and a simulation tool. The interface is capable of taking the relevant data for a given analysis from the design space model and translating it into a format which the simulation tool can work with. The interface then runs the simulations, processes the data given back to it by the simulation tool, catalogs it, and lets the user access this information. As a result, all the design space and analysis data exists within one model in a format which can be automatically translated to associated simulation tools, which eliminates the need for design teams to manually translate data from one simulation tool to another. The result is that an engineer can operate solely within the confines of the design space modeling tool, analyzing the design space in various ways and critiquing the results without even having knowledge of the existence of the separate simulation tools.

Integration of Multirate ESL solver into VTB

[13] Pearce, J.G. 2010, "Multi-rate and Integrated Package Simulation", In Proceedings of the 2010 Grand Challenges in Modeling & Simulation Conference (GCMS), (Ottawa, Canada, July 11-14). SCS, San Diego, CA, 473-479.

This paper investigates the solution of multi-rate simulation problems using the ESL simulation tool and the VTB. A brief description of ESL is presented with particular emphasis on its *segment* feature, which may be used for multi-rate problems. Using a three-network tri-rate electrical circuit example, each network is modeled as an ESL segment and alternative inter-segment coupling methods are compared. A method of

importing ESL segments into VTB entities using .NET technology is described and a solution of the multi-rate example comprising such entities within VTB is presented.

[14] Pearce, J.G., Kraft, R.J. 2011, "Multi-discipline, Multi-tool Simulation Developments", In Proceedings of the 2011 Grand Challenges in Modeling & Simulation Conference (GCMS), (The Hague, Netherlands, June 27-29). SCS, San Diego, CA. (Received GCMS Prize Paper award).

The theme of this paper is the integration of diverse simulation tools within a single unified environment to facilitate multi-discipline simulations. The example used to illustrate this is the integration of the ESL simulation tool into the VTB. Previous work involving the importation of ESL segments into early versions of VTB using COM and .NET technologies is described. The objectives and progress of the current project, aimed at achieving a greater degree of integration, are presented. In particular, the methodology behind an ESL Solver which links the ESL Simulation engine with the VTB and the creation of an extendible set of basic ESL Entities in VTB are described.

[15] Pearce J.G. and Kraft R.J., 2012, "Developments in Integration of Simulation Tools", Proceedings of the 2012 International Conference on Grand Challenges in Modeling and Simulation, Genoa, Italy, July 8-11, 2012, SCS.

This paper reports on progress of the ongoing project to integrate ESL into VTB, and concentrates on aspects of modularization. The ability to apply VTB's native subsystem facility to schematics comprising ESL entities is demonstrated. The mapping of modules, created using VTBs *Module Designer* tool, into ESL submodels, to avoid code duplication and ensure efficient execution, is described. A technique which addresses a requirement allow a user to include ESL source code in a VTB simulation by importing ESL textual submodels is presented. A new tool for the importation of such submodels *ESL Submodel to VTB Entity* is described.

Low-cost method for DSP-based simulation and control

[16] Fetzer, D., "Development and implementation of a real-time network simulation solver on a cluster of digital signal processors" Master Thesis A.

Increasing energy security by becoming less dependent on fossil fuel imports, mitigating climate change by reducing CO2 emissions while avoiding an increased use of nuclear power and considering the depletability of fossil resources are the main drivers behind the increased use of renewable energy sources (RES).

As a lot of the RES are not located near the centers of consumption one can also speak of distributed RES. The main problem of an electricity system with a high penetration of DRES and RES is that their power and energy output is mainly influenced by the weather. To overcome this problem measures as demand side management, storage, grid reinforcement and the direct control of renewable energy generators can be taken. Some of these measures might be implemented by coupling the electrical and thermal system. Altogether these measures will lead to an energy system with increased complexity compared to the systems in place today.

Testing the behavior of single components in their system environment is economically unfea- sible. Instead, hardware in the loop testing, as already widely used in the

automotive or aviation industries, will also become increasingly important in the energy sector. During hardware in the loop tests, the system behavior is tested by simulating the environment in real time and letting the simulated environment interact with the component. A detailed discussion on real time simulation and its necessity can be found in [1].

In a power system with high penetration of RES, many power electronic components will be present. In hardware in the loop tests, these components have to be simulated in real time with a very small time step due to the high switching frequencies of the converters. Thus parallelization has to be utilized because state of the art single computational units do not have the computational speed required.

Furthermore also thermal devices like machines or hot water storage tanks will have to be simulated together with the electrical system. That means that the simulation environment should be capable of doing multiphysics simulations.

From this, the need of a multiphysics real time simulation platform can be derived. At the Institute for Automation of Complex Power Systems and the Institute for Power Generation and Storage Systems at RWTH Aachen such a platform is under development and this thesis was written in this context. The overall project is now described. The real time simulation platform consists of a hardware and a software part. On the hardware side the simulator consists of multiple digital signal processor (DSP) boards which are connected in parallel. The boards are equipped with inputs and outputs for communicating with their environment. A more detailed description can be found in [2] and [3]. On the software side there is the algorithm of the network simulation, which can be uploaded as C code onto the boards. Of fundamental importance are the board drivers that, among other tasks, take care of the memory management. The drivers and the board itself are developed at the Instituted for Power Generation and Storage Systems at RWTH Aachen. The network solver as well as component models are developed at the Institute of Complex Power Systems (ACS) at RWTH Aachen.

This thesis was written in the context of the solver and component development at ACS. In more detail the resistive companion solver provided by the software called the Virtual Test Bed (VTB) was modified such that parallel simulation of a dynamic natural coupled network becomes possible. Furthermore a doubly fed induction wind generator system was modeled and implemented as a test case.

[17] M. Marin, A. Benigni, H. Lakhdar, A. Monti, P. J. Collins, "Towards the implementation of a parallel real-time simulator for DSP cluster" Proc. Grand Challenges in Modeling and Simulation, July 2012, Genoa, Italy, pp. 1-6.

This work presents the recent development in the definition, analysis and implementation of a new approach to parallel, real-time simulation of dynamic nonlinear power systems. This method combines the advantages of the Resistive Companion Method (RCM) with those of the state equation approach, in order to overcome their respective weaknesses when large systems have to be simulated in real-time.

•Methods to account for the combined effects of power flow and information flow on the performance of a system

[18] Nwankpa, C. O., Jimenez, J. C., Jayasuriya, S., "Modeling and simulation of information-embedded multi-converter power systems," in Proc. 2013 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1544-1547, May 2013.

Information-embedded multi-converter systems can be found in modern shipboard power systems, dc distribution systems and renewable energy systems. The focus of this work is the incorporation of underlying coupling issues among the converters into their modeling and simulation. This inherent cross-regulation behavior can lead to exceeding controllers' limits, which results in harmful operation situations during system disturbances. Thus the incorporation of these system dynamics into their corresponding modeling formulation is an important feature. This paper addresses the aspects of modeling and simulation of multi-converter based power systems with the inclusion of power electronic equipment dynamics in the overall system model.

[19] DeCelles, S., and Kandasamy, N., "Entropy-based Detection of Incipient Faults in Software Systems," in Proc. 18th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC), November 2012. Best student paper award.

This paper develops and validates a methodology to detect small, incipient faults in software systems. Incipient faults such as memory leaks slowly deteriorate the software's performance over time and if left undetected, the end result is usually a complete system failure. The proposed method combines tools from information theory and statistics: entropy and principal component analysis (PCA). The entropy calculation summarizes the information content associated with the collected low-level metrics and reduces the computational burden incurred by the subsequent PCA step which detects underlying patterns and correlations present in the multivariate data, as well as distortions in the correlations indicative of an incipient fault. We use the technique to detect memory bloat within the Trade6 enterprise application under dynamic workload patterns, showing that small leaks can be detected quickly and with a low false alarm rate. Our method is also robust to the periodic/seasonal patterns affecting the metrics used to detect the fault.

[20] Saichol Chudjuarjeen, Juan Jimenez, Sachi Jayasuriya, Chika Nwankpa, Karen Miu and Anawach Sangswang, "DC-DC Boost Converter with Network Model for Photovoltaic System", Proceedings of 2011 ECCE, Phoenix, AZ., September 17-22.

Power electronic converters are an important feature of photovoltaic systems. The operations of such systems are highly reliant on their embedded communication infrastructures. Communication delays in delivering converter measurements across a computer controlled network can affect the accuracy of these measurements as viewed by remote hosts. With the preceding as motivation, this paper presents the simulation of an information embedded boost converter. The embedded information network will be represented two ways; by a logistic growth and an exponential model. Results will concentrate on the effect of delays on the overall analysis of the converter.

[21] Jimenez, J.C., Jayasuriya, S., Nwankpa, C. O., "Effects of communication network and load parameters on information-embedded multi-converter shipboard power systems," in

Proc. 2013 IEEE Electric Ship Technologies Symposium (ESTS), pp. 320-325, April 2013.

Information-embedded multi-converter shipboard power systems are comprised of multiple independently operated converters that work together with electromechanical equipment. Their relations can produce a large variety of dynamic and static interactions that can lead to irregular behavior of a converter, a group of converters or the whole system. This inherent cross-regulation behavior may correspond to controller limit violations, which results in harmful operational situations during system disturbances. It is significant then to monitor the system during perturbations; their underlying dynamics exhibit trajectories that not only depend on their independent operation modes. Appropriate modeling is a key issue to power system analysis and warrants primary investigation. This paper addresses aspects of modeling and simulation of multi-converter based shipboard power systems with the inclusion of power electronic equipment dynamics and network control in the overall system model. The model behavior is evaluated with respect to changes in loading conditions and control parameters (local and network) and validated through simulation of a nonlinear observability formulation of the developed system model.

[22] Saichol Chudjuarjeen, Sachi Jayasuriya, Samart Yachiangkam, Juan C. Jimenez, Chika O. Nwankpa, Karen Miu and Anawach Sangswang, "Analysis of Measurement Delay Errors in a DC-DC Buck-Boost Converter using Stochastic Differential Equations," Proceedings of IECON 2011, Melbourne, Australia.

This paper presents the use of stochastic differential equations (SDEs) to account for effects of network traffic on the communication of measurement data from a DC-DC Buck-Boost converter to a remote control center. The work is motivated by the need to account for observability of system states when a converter or system of converters is remotely monitored through a communication network comprised of switched networks. This work builds upon literature on SDEs used as appropriate tools to quantify the effects of uncertainty reflected in switched networks alone. Saichol Chudjuarjeen, Sachi Jayasuriya, Juan C. Jimenez, Chika O. Nwankpa, Karen Miu and Anawach Sangswang, "Simulation of A DC-DC Boost Converter with Network Models" Proceedings of 2011 Summer Simulation Multiconference, Den Haag, The Netherlands, June 27-29.

This paper presents a simulation tool used for the identification of regions of operation and deterministic analysis of a DC-DC boost converter for varying system parameters. Here the focus is on a pulsed-width modulated (PWM) dc-dc boost converter with feedback control and network models used to represent the delay experienced by measurements due to network traffic.